

**AMENDMENTS TO THE DRAWINGS**

The attached sheet of drawings includes changes to Figure 2. This sheet replaces the original sheet of Figure 2. Figure 2 has been amended to conform more clearly with its corresponding detailed description. Specifically, the figure has been designated as Prior Art.

Attachment: Replacement Drawing Sheet

### REMARKS

Claims 27, 29-42, 44-46, 48-50 and 52-54 are pending in the application.

Claims 27, 29-42, 44-46, 48-50 and 52-54 have been rejected.

Claims 42 and 44 have been amended. No new matter has been added. Support for the amendment to claim 44 can be found, at least, in paragraphs 68-69 of the specification.

### Claim Objections

The Office Action objected to claims 42, 44, 46, and 49 as not being positively recited. In an effort to reduce confusion, the Applicants have removed the modifiers before each “means” element of the means plus function claims 42 and 44. The Applicants further note that each of these means plus function claims quite clearly recites one or more means for performing a particular function (e.g., the function of “operating upon a given location” in claim 42 and the function of “obtaining a designation” in claim 44). Accordingly, these two claims quite clearly recite a positive means for performing a function.

The Office Action also objected to system claims 46 and 49 as not being positively recited. The Office Action appears to be stating that the phrase ‘configured to’ is not a positive recitation, because the system includes a module “with only intention and capable [sic] of performing the operation.” Office Action mailed June 20, 2007 (hereinafter “OA”), p. 3. However, the Applicants note that the use of “configured to” is appropriate and in no way fails to “positively recite” the functionality that is provided by the system of claims 46 and 49. “There is nothing inherently wrong with defining some part of an invention in functional terms. Functional language does not, in and of itself, render a claim improper.” MPEP § 2173.05(g), citing *In re Swinehart*, 439 F.2d 210, 169 USPQ 226 (CCPA 1971). Furthermore, in cases such as *State Street Bank & Trust Co. v. Signature Financial Group Inc.*, claims have been found valid despite their use of the language “configured to.” 47 USPQ2d 1596 (Fed. Cir. 1998). Accordingly, the Applicants respectfully request the withdrawal of this objection.

### Rejection of Claims under 35 U.S.C. §101

Claims 42-49 stand rejected under 35 U.S.C. §101 as purportedly being directed to non-statutory subject matter. The Applicants have amended claim 42 to explicitly

recite a means for storing a sieve. Such a means for storing (e.g., a memory or other storage device) is clearly not software per se. Accordingly, the Applications respectfully request the withdrawal of this rejection.

*Rejection of Claims under 35 U.S.C. §102(e)*

Claims 27, 29-30, 32-42, 44-46, 48-50 and 52-54 stand rejected under 35 U.S.C. §102(e) as being anticipated by Miyata et al. (USPPN 2003/0225972) hereinafter referred to as (“Miyata”). Applicants respectfully traverse this rejection.

With respect to claim 27, the cited art fails to anticipate, teach, or suggest “a request to perform an operation on a first set of locations of a plurality of locations in a storage area.” The Final Office Action mailed January 31, 2007 (hereinafter referred to as “FOA”) cites FIG. 11 and paragraphs 87-88 of Miyata as teaching this feature, stating that Miyata teaches how to: “perform a copy duplicate operation in response to a request from client computer.” FOA, p. 3.

FIG. 11 does not show or suggest any request to perform an operation on a particular set of locations in a storage area. Instead, “FIG. 11 is a flowchart for registering/resetting the duplicating function.” Miyata, paragraph 87. Paragraphs 87-88 describe FIG. 11. These paragraphs describe how the method illustrated in FIG. 11 determines a file or directory for registering/resetting the duplicating function. The method of FIG. 11 does not appear to be performed in response to a request, nor does this method appear to be performed on a set of locations.

Furthermore, since the registering/resetting of FIG. 11 is performed for files and directories (as opposed to being performed on data at a location within a storage area), it seems quite unlikely that the method FIG. 11 would be performed in response to a request like the one specified in claim 27. Operating on a file or directory is quite different than operating on a location within a storage area; the former operates on a logical construct that can be located in any of a variety of possible physical locations, while the latter operates on a specific location within a storage area, regardless of whether that specific location stores data for a particular file or directory. Claim 27 describes a request that an operation be performed on a particular set of locations (as opposed to requesting that an operation be performed on a particular file or directory). Thus, no request to perform an operation on locations of a storage device is taught or

suggested in this section of Miyata, which describes performing registering/resetting for files or directories. For at least this reason, Miyata does not anticipate claim 27.

Nevertheless, the Examiner states that Miyata's caching the contents of a data block (as described in paragraph 81 of Miyata) anticipates "the claim's operations on locations in storage area." FOA, p. 3. Applicants first note that paragraph 81, which describes FIG. 9 of Miyata, appears to have no relationship whatsoever to the actions described in the portions of Miyata described above. Furthermore, nothing in paragraph 81 describes a request to perform an operation on a set of storage locations. Instead, paragraph 81 describes instructing a storage unit to set an attribute bit in order to indicate that a block is allowed to reside in cache. This instruction, at best, instructs the storage unit to modify a single bit, not to perform an operation on a set of storage locations. Accordingly, this portion of Miyata also fails to teach or suggest the above-quoted feature of claim 27.

Additionally, the cited art does not teach or suggest "performing the operation upon a given location in the first set of locations of the plurality of locations in the storage area only if the given location is described in at least one location description of a sieve associated with the operation, wherein the sieve comprises the at least one location description and a corresponding property, wherein the property comprises information identifying the operation, and wherein the at least one location description identifies the only storage locations within the storage area upon which the operation can be performed," as recited in claim 27. In other words, the cited art does not teach or suggest performing an operation on a location only if the sieve that describes the operation identifies the location as being one of the only locations within the storage area upon which the operation can be performed.

The FOA equates the metadata structure of FIG. 4 of Miyata with the sieve of claim 27. FOA, p. 4. This metadata structure includes a file attribute array region 402, which the Examiner equates each individual attribute bit in this array with both the location description and the property of claim 27. FOA, p. 4.

Applicants first note that the Examiner appears to be confusing the block attribute bit 203 of FIG. 2 of Miyata (which was relied on to anticipate the location description of claim 27 in prior Office Actions) with the file attribute array region 402 of FIG. 4 of

Miyata. The block attribute bit of Miyata is a bit that is associated with a block of a storage device. Miyata, paragraphs 41-43. The file attribute array region of Miyata is a region that stores metadata for each file within a file system. Miyata, paragraphs 52-53. The file attribute array region does not provide any information that corresponds to either the property or the location description of claim 27, and thus this element of Miyata clearly neither teaches nor suggests the sieve of claim 27.

Furthermore, the block attribute bit of Miyata also fails to anticipate the sieve of claim 27. Each block attribute bit represents an attribute and is part of an attribute field. Miyata, paragraph 42. As shown in FIG. 2, each attribute field is associated with a single logical block address (LBA). Accordingly, each block attribute field is associated with a single LBA and identifies the attribute(s) of that LBA. Since each block attribute field represents only a single LBA and since different block attribute fields can contain the same bit sequences (e.g., as shown by the block attribute bit fields of LBAs 1-4 of FIG. 2 having the same value), Miyata's system allows multiple block attribute fields to identify their corresponding LBAs as having the same attributes. Accordingly, a given block attribute field clearly does not identify all locations that have the same attribute. Thus, a single block attribute bit, or even a single block attribute field clearly cannot identify all locations that have same attribute; instead, each such bit or field only (indirectly) identifies a single LBA. Furthermore, the cited portions of Miyata indicate that Miyata's block attribute field identifies attributes, not operations. Accordingly, Miyata's system similarly does not teach or suggest a location description like that recited in amended claim 27, which identifies all of the storage locations within a storage area upon which an operation can be performed.

In response to similar arguments, the Examiner states that the file attribute array region 402 can identify multiple LBAs within a given file. FOA, pp. 8-9. As noted above, however, the file attribute array region 402 has nothing to do with the block attribute bits, nor does the file attribute array region include the same information as the sieve of claim 27. Accordingly, it is irrelevant whether the file attribute array region can identify multiple LBAs.

Furthermore, the cited portions of Miyata clearly do not teach or suggest not a sieve like that recited in amended claim 27. In particular, Miyata does not teach or suggest a sieve that is associated with an operation and that includes both a property,

which includes information that identifies the operation, and a location description that identifies all locations within a storage area upon which the operation (as identified by the property) can be performed. As noted above, Miyata's attribute bit is only associated with a single LBA, and thus cannot identify all locations upon which an operation can be performed. Furthermore, while the attribute bit can be associated with an operation (e.g., the ditto operation described in Miyata), the attribute bit quite clearly does not include information that identifies an operation. Accordingly, the portions of Miyata describing the attribute bit clearly neither teach nor suggest the sieve of claim 27.

In response the above arguments, the Examiner first states that Miyata's duplicate function is "formed/registered in response to a request for duplication operation." OA, page 9. However, the Examiner has not provided any support in the cited art for this position, nor are the Applicants able to find such support in the cited portions of Miyata. Accordingly, the Applicants respectfully submit that the cited portions of Miyata fail to teach or suggest the "request" of claim 27.

Furthermore, the Examiner states that "Miyata's [sic] clearly discloses that the operation is performed for set of locations." OA, p. 10. The Applicants note that claim 27 does not recite an operation that is performed for a set of locations (instead, claim 27 recites a request to perform an operation on a first set of locations), and thus it is irrelevant whether Miyata teaches such an operation, since Miyata clearly fails to teach or suggest a request to perform such an operation.

Furthermore, the Applicants again note that the Examiner has still not shown that the operation is performed upon a given location within the first set "only if the given location is identified in at least one location description of a sieve associated with the operation." Finally, the Examiner has not shown how the cited art teaches or suggests a sieve that includes a property, which "comprises information identifying the operation." As noted above, Miyata's attribute bit can be associated with an operation but does not include information that identifies an operation.

Accordingly, Applicants submit that claim 27 is patentable over the cited art. Applicants submit that these arguments apply with equal force to independent claims 42, 46, and 50. Thus, independent claims 27, 42, 46, and 50 are allowable for at least the foregoing reasons. The respective dependent claims 29-30, 32-41, 43-45, 47-49, 51-54,

and 58-61 are allowable for at least the same reasons that claims 27, 42, 46, and 50 are allowable.

*Rejection of Claims under 35 U.S.C. §103(a)*

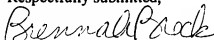
Claim 31 is rejected under 35 U.S.C. § 103(a) as being unpatentable over Miyata as applied to claim 27, and further in view of Krishnamurthy (U.S. Patent No. 6,823,436), hereinafter referred to as ("Krishnamurthy"). Applicants respectfully traverse this rejection for at least the foregoing reasons presented above.

**CONCLUSION**

In view of the amendments and remarks set forth herein, the application and the claims therein are believed to be in condition for allowance without any further examination and a notice to that effect is solicited. Nonetheless, should any issues remain that might be subject to resolution through a telephone interview, the Examiner is invited to telephone the undersigned at 512-439-5087.

If any extensions of time under 37 C.F.R. § 1.136(a) are required in order for this submission to be considered timely, Applicant hereby petitions for such extensions. Applicant also hereby authorizes that any fees due for such extensions or any other fee associated with this submission, as specified in 37 C.F.R. § 1.16 or § 1.17, be charged to deposit account 502306.

Respectfully submitted,



Brenna A. Brock  
Attorney for Applicants  
Reg. No. 48,509  
(512) 439-5087 [Phone]  
(512) 439-5099 [Fax]